BokodeVision: Gaze-Activated Bokode Display for Smart Glass Interactions

Pranava Singhal

Abstract—Current interaction methods for smart glasses, such as voice commands, touch controls, and hand gestures suffer from key limitations like social awkwardness, fatigue, and low accuracy. This project explores gaze as an alternative interaction mechanism for smart glasses. It focuses on building a near-eye display using Bokodes and integrating it with an off-the-shelf eye tracker from Pupil Labs. The display is controlled by gaze-based commands, offering a more intuitive, effortless, and efficient interaction method. All code and design files for this project can be found at https://github.com/pringlesinghal/BokodeVision.

Index Terms—Bokode, Gaze Tracking, Near-eye Display, Smart Glasses

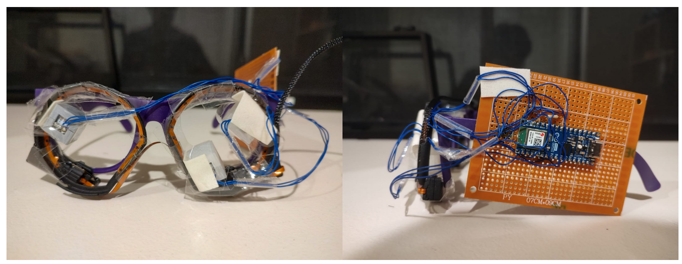
1 Introduction

Smart glasses are rapidly emerging as disruptive technology, with a projected market size of 1.93 billion by 2024 [1], reflecting their increasing adoption in both the industrial and consumer sectors. Despite their growing popularity, current smart glasses mainly rely on interaction methods that have significant drawbacks.

- **Voice Commands**:
  - *Social Awkwardness*: Voice commands can cause social awkwardness when used in public settings [2].
  - *Noise Sensitivity*: They suffer from reduced accuracy in noisy environments.
- **Touch Controls**:
  - *Slower Task Completion*: Touch controls result in slower task completion times.
  - *Discomfort*: They can cause discomfort due to prolonged usage [3].
- **Hand Gestures**:
  - *Low Accuracy*: Hand gestures are plagued by low accuracy.
  - *Fatigue*: They can induce fatigue even after short periods of use [4].

The limitations of current interaction methods for smart glasses underscore the need for alternative solutions. Gaze-based interaction presents a promising alternative, offering unobtrusiveness, ease, and accuracy. Using gaze as a primary interaction mechanism, smart glasses can address the social, accuracy, and usability challenges inherent in traditional methods. This project investigates the feasibility of integrating gaze-based interaction into smart glasses, focusing on a near-eye display that can be activated and controlled by the user's gaze, thereby enhancing the interaction experience.

P. Singhal is with the Department of Computer Science, Stanford University, Stanford, CA.
E-mail: psinghal@stanford.edu



(a) Front View (b) Side View

Fig. 1. Views of the Smart Glasses Prototype.

2 Related Work

2.1 Gaze-Tracking in Smart Glasses

The use of gaze-tracking within smart glasses is not new. GazeGPT [5] explored the use of gaze cues for improving spatial localization in vision-language model queries. They track eye movement over the entire field-of-view and create multi scale crops of the camera captures before prompting a vision language model with the user's query in order to ensure more accurate and relevant responses.

Gaze tracking is also commonly used in Virtual Reality (VR) headsets in order to create gaze-based selection UI such as controlling the cursor or navigating menus. It has also enabled more efficient foveated rendering [6].

2.2 Near-Eye Displays

Previous smart glass designs have looked at always-on displays that are placed on the glasses frame and project content into the users eyes. [7] designed a near-eye display using computer-generated holograms of symbols that are lit by a coherent source such as lasers and are placed on the upper rim of the glass. These are turned on based on the user's activity. The new Halliday glasses [8] also feature a compact display that shows complex information such as UI menus or text translations depending on the task at hand. These displays are designed to be private and compact.

3 Proposed Method

Our hypothesis is that previous works using near-eye displays are typically always on or activated based on the task which is for a prolonged duration. Continuous display can cause eye fatigue. We believe that a gaze-activated display is a natural choice to reduce the total activation time of the display.

We also hypothesize that user's rarely look into their peripheral vision which makes it a natural choice for placing the gaze-activated display. This way day-to-day activities will not unnecessarily turn on the display unless the user conciously tries to activate it.

Existing gaze-tracking designs in smart glasses track gaze over the entire field-of-view (FoV). While this is useful for several applications, it requires expensive camera-based tracking with computationally expensive image processing algorithms. Given the limited computational budget on smart glasses with a sleek form factor we seek to explore computationally inexpensive gaze tracking designs which can reduce the processing overhead. This is made possible since we only need to detect whether a user is looking at the display or not which is far simpler than tracking gaze over the entire FoV.

3.1 Gaze-Activation as Feedback UI

Our display can serve as visual feedback for the user. For instance, if one wants to activate voice assistant on the smart glasses, they can simply gaze at the top right corner of their glasses where a small microphone icon will be displayed showing the user that the voice agent is activated. The display will turn off as soon as the user's gaze returns from the periphery.

Other examples could be picking a phone call, entering maps navigation through in-ear voice guidance, or taking a picture with the on-frame camera. In each case a phone icon, maps icon, or camera icon can give the user feedback to know which action has been activated. This is more important when there are multiple gaze-activated displays located at different points on the periphery.

3.2 Optical System

The optical system is inspired by Bokodes [9] which consist of a small lens, an optical transparency with patterns, a diffuser and light source. While the original design was not intended for near eye displays but rather as a compact alternative to barcodes, we repurpose the optical system of Bokodes for our near-eye display.

3.3 Gaze Tracker Design and Next Steps

We use an open-source Pupil Labs eye tracker for this prototype. This is a camera base eye-tracker. Since we do not have a world camera in this case, we write our own gaze tracking scripts which calibrate pupil position with a display activation sequence. We use Mahalanobis distance between pupil coordinates and estimated display coordinates from calibration for robust tracking. We additionally implemented hysteresis, saccade rejection, and a minimum fixation time to reduce flickering of the display.

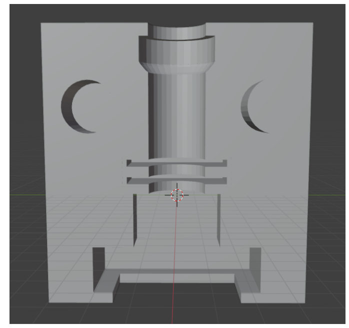


Fig. 2. Front-View of 3D model

In the long-run the goal is to switch to infra-red and photodiode based gaze detection which can be more compact, computationally efficient, and light weight. The idea is to use the corneal reflectance properties, especially the fact that IR reflection peaks when the user is directly looking at the IR source. We did not pursue this design due to safety concerns with near-eye IR light source and plan to proceed in this direction only after more safety measurements and testing.

4 Experimental Results

4.1 Optical System

4.1.1 3D Models of Optical Housing

I designed a custom housing for the optical subsystem in Blender to accomodate the lens, transparency, diffuser, and light source. This was the most challenging and time consuming component of this project. The system was designed by first creating the light path including space for components and then subtracting that from a cuboid. I designed cylindrical connectors to make it a press fit.

The designs were printed with the Prusa MK3.9 3D printers in Lab64 in Packard building in Stanford University. It took more than 20 prints to arrive at the final design. The final design satisfied the following criteria:

- The lens and LED was supported from both sides and did not slide
- The optical transparency and diffuser (sheet of tracing paper) were supported on both sides and did not slide
- The optical housing could be closed after inserting the lens, transparency, diffuser, and LED
- The spacing between the lens and transparency was optimal for the user to see the symbol clearly
- The transparency and diffuser sheet did not get crumpled when the optical housing was closed
- The two pieces of the optical housing would tightly fit together when pressed and not get separated unless some force was applied. However, the two pieces should not be completely stuck. This requires choosing an optimal ratio for the inner and outer cylinder radii.

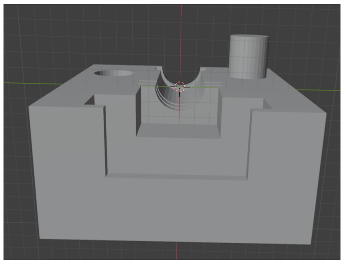


Fig. 3. Side-View of 3D model

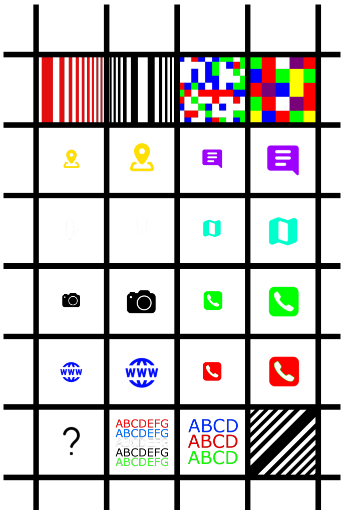


Fig. 4. 5x5mm Icons for Near-Eye ”Bokode” display transparency slide

4.1.2 *UI Symbols for Prototyping*

I designed diverse UI icon symbols with GIMP and had them printed on 5x5mm squares on an e6 optical transparency that was printed at the Oscar Photo Lab in San Francisco. These are shown in the figure below. I also made some test patterns besides icons such as chirp symbols and colour blocks to see how the display and prints are in terms of resolution and colour accuracy. I also printed text considering future expansions with more complex text feedback in near eye displays such as the Halliday glasses [8].

4.1.3 *Gaze Tracking Prototype*

I used the HTC Vive add-on by Pupil Labs for eye-tracking. While this is originally meant to be used with the HTC Vive headset, I repurposed to use it as a standalone device. Using the open source Pupil Labs API for python I was able to stream real-time pupil coordinates.

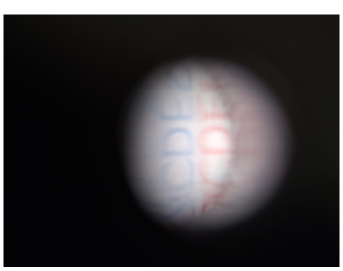


Fig. 5. Text example in Bokode Display: This was captured through a mobile camera by adjusting the focus to infinity since the plano-convex lens in the Bokode projects the transparency placed at focus to infinity

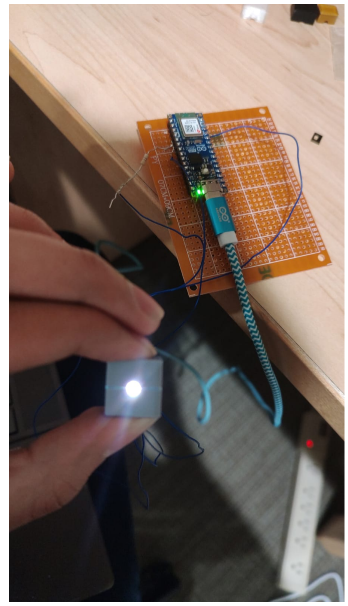


Fig. 6. Initial circuit setup for testing of the optical housing and Bokode display

The calibration procedure consisted of the near-eye displays turning on one-by-one prompting the user to look at the active display for 3 seconds. In the duration we are continuously recording the pupil coordinates and compute the mean vector and covariance matrix of coordinates. After calibration is complete for all displays, we start tracking the users pupil and measure the Mahalanobis (covariance weighted) distance between the temporally-smoothed pupil coordinates and the display and record an activation signal if the pupil is within a certain distance threshold. Moreover, we also add a hysteresis state machine to count the number of activations of deactivations before transitioning between on and off states. We also measure pupil velocity and ignore activation signals if the velocity exceeds a threshold (saccade detection). We detect fixations by ensuring a minimum duration of 400ms for which the user is constantly looking at the display (activation signals are triggered for this duration). Using this criteria we are able to create a much smoother gaze based display activation experience than was possible without.

5 Conclusion and Future Work

The next step would be to explore how near-IR and photodiode based eye-tracking can be integrated into the design both in the optical housing and also from an algorithmic standpoint for processing the photodiode readings to detect a peak (user gaze). Moreover, we can explore the possibility of patenting this new interaction mechanism for smart glasses.

Acknowledgments

I would like to thank Professor Gordon Wetzstein for his invaluable mentorship and the Electrical Engineering Department at Stanford University for sponsoring the development of this prototype. I would also like to thank Jeff Stribling from Lab64 for his guidance with making the 3D prints for my optical housing. I'm grateful to Neil from the PupilLabs team for helping be connect to the eye-tracker by directing me to the relevant sections of the PupilCore API.

References

[1] "Smart glasses market size, share & trends analysis report," 2024. [Online]. Available: https://www.grandviewresearch.com/industry-analysis/smart-glasses-market-report

[2] A. Easwara Moorthy and K. P. L. Vu, "Privacy concerns for use of voice activated personal assistant in the public space," *International Journal of Human–Computer Interaction*, vol. 31, no. 4, pp. 307–335, 2015.

[3] Y. Hsieh, A. Jylhä, V. Orso, L. Gamberini, and G. Jacucci, "Designing a willing-to-use-in-public hand gestural interaction technique for smart glasses," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 2016.

[4] N. Mohamed, M. Mustafa, and N. Jomhari, "A review of the hand gesture recognition system: Current progress and future directions," *IEEE Access*, pp. 1–1, 2021.

[5] R. Konrad, N. Padmanaban, J. Buckmaster, K. Boyle, and G. Wetzstein, "Gazegpt: Augmenting human capabilities using gaze-contingent contextual ai for smart eyewear," *arXiv*, 2024.

[6] A. Patney, M. Salvi, J. Kim, A. Kaplanyan, C. Wyman, N. Benty, D. P. Luebke, and A. E. Lefohn, "Towards foveated rendering for gaze-tracked virtual reality," *ACM Transactions on Graphics (TOG)*, vol. 35, pp. 1 – 12, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:17614452

[7] A. Olwal and B. Kress, "1d eyewear: Peripheral, hidden leds and near-eye holographic displays for unobtrusive augmentation," in *Proceedings of the 2018 ACM International Symposium on Wearable Computers*, 2018.

[8] "Halliday global website," 2024, accessed: 2024-03-14. [Online]. Available: https://hallidayglobal.com/

[9] A. Mohan, G. Woo, S. Hiura, Q. Smithwick, and R. Raskar, "Bokode: Imperceptible visual tags for camera-based interaction from a distance," in *ACM SIGGRAPH 2009 papers*, 2009.

Pranava Singhal Pranava is a first-year Master's in Computer Science student. He has recently developed a keen interest in the XR space through Professor Wetzstein's EE367 class and XR hackathons like Immerse the Bay and MIT Reality Hack. His previous research has been focused on machine learning and signal processing topics like density estimation, federated learning and causal inference. He has published at ICML, NeurIPS, EUSIPCO, and IEEE Networking Letters.